

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-326935

(43)Date of publication of application : 22.11.2001

(51)Int.Cl.

H04N 7/30

H03M 7/30

(21)Application number : 2000-141675

(71)Applicant : HUDSON SOFT CO LTD

(22)Date of filing : 15.05.2000

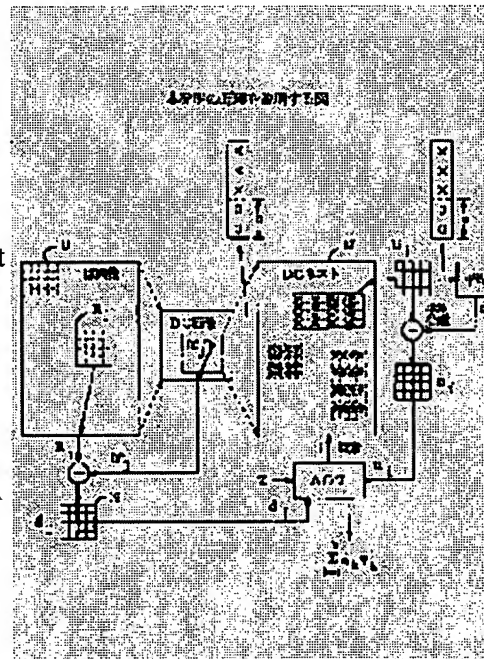
(72)Inventor : ITAGAKI FUMIHIKO  
KAWASHIMA MIYUKI

## (54) IMAGE CODING/DECODING METHOD, ITS DEVICE, AND RECORDING MEDIUM RECORDED WITH PROGRAM THEREFOR

(57)Abstract:

PROBLEM TO BE SOLVED: To obtain high image quality and high-speed coding/decoding in an image coding/decoding method, its device, and a recording medium in which its program is recorded.

SOLUTION: In an image coding method to generate a DC image consisting of each block average obtained by dividing an image data into blocks for every B pixel for making the part into a DC nest, and to obtain one or two or more orthogonal bases ( $\alpha_k$   $\langle vk \rangle$  or the like) for approximating the residual vector  $\langle dj \rangle$  by an accommodative orthogonal transformation (AOT) using DC nest when the size of a residual vector  $\langle dj \rangle$  obtained by separating a DC value DCJ from a pixel block  $\langle Rj \rangle$  for coding exceeds an allowable value Z, each inferior n bit ( $n = \log_2 B$ ) of a base candidate block  $\langle Ui \rangle$  to which a down sample is applied from the DC nest is set as 0. Besides, the block average  $ai$  is separated from the base candidate block  $\langle Ui \rangle$  to generate a base candidate vector  $\langle ui \rangle$ .



(19) 日本国特許庁 (J P)

## (12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-326935

(P2001-326935A)

(43) 公開日 平成13年11月22日 (2001.11.22)

(51)Int.Cl. <sup>7</sup>	識別記号	F I	テーマコード(参考)		
H 0 4 N	7/30	H 0 3 M	7/30	A	5 C 0 5 9
H 0 3 M	7/30			B	5 J 0 6 4
		H 0 4 N	7/133	Z	

審査請求 未請求 請求項の数16 O L (全 33 頁)

(21) 出願番号	特願2000-141675(P2000-141675)	(71) 出願人	591095856 株式会社ハドソン 北海道札幌市豊平区平岸三条五丁目4番22号
(22) 出願日	平成12年5月15日 (2000.5.15)	(72) 発明者	板垣 史彦 北海道札幌市南区芸術の森3丁目C62 株式会社ハドソン内
		(72) 発明者	川島 深雪 北海道札幌市南区芸術の森3丁目C62 株式会社ハドソン内
		(74) 代理人	100097087 弁理士 ▲高▼須 宏

最終頁に続く

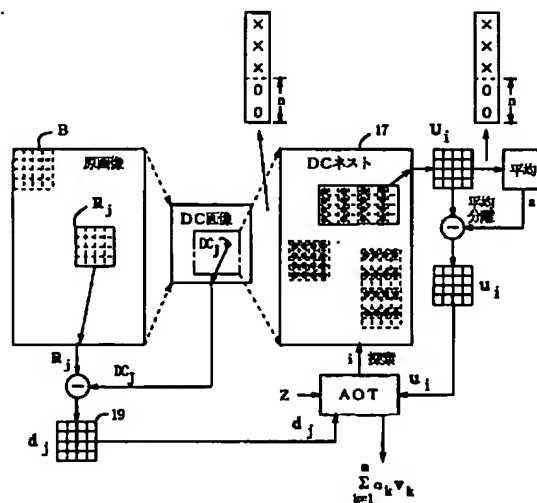
(54) 【発明の名称】 画像符号/復号方法及びその装置並びにそのプログラムを記録した記録媒体

(57) 【要約】

【課題】 画像符号/復号方法及びその装置並びにそのプログラムを記録した記録媒体に関し、高画質かつ高速の符号/復号が得られることを課題とする。

【解決手段】 画像データをB画素毎にブロック分割して各ブロック平均値からなるDC画像を生成し、その一部をDCネストとすると共に、符号対象の画素ブロック  $\langle R_j \rangle$  からそのDC値  $DC_j$  を分離した残差ベクトル  $\langle d_j \rangle$  の大きさが許容値  $Z$  を超える場合は、残差ベクトル  $\langle d_j \rangle$  を近似するための1又は2以上の直交基底  $\langle \alpha_k \langle v_k \rangle \rangle$  等をDCネストを使用した適応的直交変換 (AOT) により求める画像符号方法において、DCネストからダウンサンプルされた基底候補ブロック  $\langle U_i \rangle$  の各下位  $n$  ( $n = \log_2 B$ ) ビットが0にされている。またこの基底候補ブロック  $\langle U_i \rangle$  からそのブロック平均値  $a_i$  を分離して基底候補ベクトル  $\langle u_i \rangle$  を生成する。

本発明の原理を説明する図



【特許請求の範囲】

【請求項1】 画像データをB画素毎にブロック分割して各ブロック平均値からなるDC画像を生成し、その一部をDCネストとすると共に、符号対象の画素ブロックからそのDC値を分離した残差ベクトルの大きさが許容値を超える場合は、該残差ベクトルを近似するための1又は2以上の直交基底をDCネストを使用した適応的直交変換により求める画像符号方法において、DCネストから基底候補ブロックをダウンサンプルしてそのブロック平均値を求める際の各サンプルDC画素の下位 $n$  ( $n = \log_2 B$ ) ビットが0にされていることを特徴とする画像符号方法。

【請求項2】 DC画像からDCネストを生成する際に各DC画素の下位 $n$  ビットを0にすることを特徴とする請求項1に記載の画像符号方法。

【請求項3】 下位 $n$  ビットが0にされている基底候補ブロックからそのブロック平均値を分離して残差ベクトルを近似するための基底候補ベクトルを生成することを特徴とする請求項1又は2に記載の画像符号方法。

【請求項4】 基底候補ベクトル  $\langle u_i \rangle$  の任意要素 (例えば  $u_{16}$ ) を残りの要素の一次結合で置き換えると共に、該基底候補ベクトル  $\langle u_i \rangle$  と任意他のベクトル  $\langle w \rangle$  との内積を、

$$\langle w \cdot u_i \rangle = (w_1 - w_{16}) u_1 + (w_2 - w_{16}) u_2 + \dots + (w_{15} - w_{16}) u_{15}$$

の積和演算により求めることを特徴とする請求項3に記載の画像符号方法。

【請求項5】 残差ベクトル  $\langle d \rangle$ 、基底候補ベクトル  $\langle u_i \rangle$  とするときに第1基底の探索は、
$$h_1 = \langle d \cdot u_i \rangle^2 / \|u_i\|^2$$
 を最大とするものを条件に探索することを特徴とする請求項3又は4に記載の画像符号方法。

【請求項6】 残差ベクトル  $\langle d \rangle$ 、第1基底に対応する基底候補ベクトル  $\langle u_1 \rangle$ 、第2基底を探索する基底候補ベクトル  $\langle u_i \rangle$  とするときに第2基底の探索は、
$$h_i = \{ \langle d \cdot u_i \rangle - (\langle d \cdot u_1 \rangle \langle u_1 \cdot u_i \rangle) / \|u_1\|^2 \}^2 / \{ \|u_i\|^2 - (\langle u_1 \cdot u_i \rangle)^2 / \|u_1\|^2 \}$$

を最大とするものを条件に探索することを特徴とする請求項3又は4に記載の画像符号方法。

【請求項7】 残差ベクトル  $\langle d \rangle$ 、第1正規基底ベクトル  $\langle v_1 \rangle$ 、第2正規直交基底ベクトル  $\langle v_2 \rangle$ 、第3基底を探索する基底候補ベクトル  $\langle u_i \rangle$  とするときに第3基底の探索は、

$$h_i = \{ \langle d \cdot u_i \rangle - (\langle d \cdot v_1 \rangle \langle v_1 \cdot u_i \rangle - \langle d \cdot v_2 \rangle \langle v_2 \cdot u_i \rangle) / \{ \|u_i\|^2 - \langle v_1 \cdot u_i \rangle^2 - \langle v_2 \cdot u_i \rangle^2 \} \}^2 / \{ \|u_i\|^2 - \langle v_1 \cdot u_i \rangle^2 - \langle v_2 \cdot u_i \rangle^2 \}$$

を最大とするものを条件に探索することを特徴とする請求項3又は4に記載の画像符号方法。

【請求項8】 探索条件にマッチした基底候補ベクトル

$\langle u_i \rangle$  をそれ以前の1又は2以上の正規直交基底に正規直交化することを特徴とする請求項6又は7に記載の画像符号方法。

【請求項9】 画像データをB画素毎にブロック分割して各ブロック平均値からなるDC画像を生成し、その一部をDCネストとすると共に、符号対象の画素ブロックからそのDC値を分離した残差ベクトルの大きさが許容値を超える場合は、該残差ベクトルを近似するための1又は2以上の直交基底をDCネストを使用した適応的直交変換により求める画像符号方法において、求められた基底系を  $\beta_k \langle u_k \rangle$  ( $k = 1 \sim m$ ) とするときに、各スカラー展開係数  $\beta_1 \sim \beta_m$  のノルムをその大きさに順に並べ替え、0を含む隣接ノルム間の各差分を求め、得られた各差分につきハフマン符号を適用することを特徴とする画像符号方法。

【請求項10】 画像データをB画素毎にブロック分割して各ブロック平均値からなるDC画像を生成し、その一部をDCネストとすると共に、符号対象の画素ブロックからそのDC値を分離した残差ベクトルの大きさが許容値を超える場合は、該残差ベクトルを近似するための1又は2以上の直交基底をDCネストを使用した適応的直交変換により求める画像符号方法において、求めた基底数が所定以上の場合は基底系の符号化に代えて、符号対象ブロックの画像データそのものを符号化することを特徴とする画像符号方法。

【請求項11】 HVQ方式に係る符号データからB画素毎の各ブロック平均値に相当するDC画像を再生し、その一部をDCネストとすると共に、ターゲットブロックのDC値に対し、前記符号データに基づきDCネストから選択生成した1又は2以上の基底ベクトルを合成してターゲットブロックの画像データを再生する画像復号方法において、

DCネストから選択ブロックをダウンサンプルしてそのブロック平均値を求める際の各サンプルDC画素の下位 $n$  ( $n = \log_2 B$ ) ビットが0にされていることを特徴とする画像復号方法。

【請求項12】 HVQ方式に係る符号データからB画素毎の各ブロック平均値に相当するDC画像を再生し、その一部をDCネストとすると共に、ターゲットブロックのDC値に対し、前記符号データに基づきDCネストから選択生成した1又は2以上の基底ベクトルを合成してターゲットブロックの画像データを再生する画像復号方法において、

復号された基底系が  $\beta_k \langle u_k \rangle$  ( $k = 1 \sim m$ ) に係る情報であるときに、DCネストから読み出した各選択ブロック  $\langle U_k \rangle$  につき各DC画素の下位 $n$  ( $n = \log_2 B$ ) ビットが0にされていると共に、まず  $\beta_k \langle U_k \rangle$  ( $k = 1 \sim m$ ) の積和演算を行い、その演算結果をブロック画素数Bで除算することを特徴とする画像復号方法。

【請求項13】 DC画像からDCネストを生成する際

に各DC画素の下位 $n$ ビットを0にすることを特徴とする請求項11又は12に記載の画像復号方法。

【請求項14】 画像データをB画素毎にブロック分割して各ブロック平均値からなるDC画像を生成し、その一部をDCネストとすると共に、符号対象の画素ブロックからそのDC値を分離した残差ベクトルの大きさが許容値を超える場合は、該残差ベクトルを近似するための1又は2以上の直交基底をDCネストを使用した適応的直交変換により求める画像符号装置において、各DCネスト画素の下位 $n$  ( $n = \log_2 B$ ) ビットが0にされているDCネストを記憶するメモリを備えることを特徴とする画像符号装置。

【請求項15】 HVQ方式に係る符号データからB画素毎の各ブロック平均値に相当するDC画像を再生し、その一部をDCネストとすると共に、ターゲットブロックのDC値に対し、前記符号データに基づきDCネストから選択生成した1又は2以上の基底ベクトルを合成してターゲットブロックの画像データを再生する画像復号装置において、各DCネスト画素の下位 $n$  ( $n = \log_2 B$ ) ビットが0にされているDCネストを記憶するメモリを備えることを特徴とする画像復号装置。

【請求項16】 請求項1乃至13の何れか1つに記載の処理をコンピュータに実行させるためのプログラムを記録したコンピュータ読取り可能な記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は画像符号／復号方法及びその装置並びにそのプログラムを記録した記録媒体に関し、更に詳しくはハイブリッドベクトル量子化(HVQ: Hybrid Vector Quantization)方式による画像符号／復号方法及びその装置並びにそのプログラムを記録した記録媒体に関する。

【0002】 今日、静止画圧縮の国際標準であるJPEG (Joint Photographic Expert Group) 方式では、 $8 \times 8$ の画素ブロックを2次元DCTによりDC値及び基本 $\sim 63$ 倍周波数の各係数値に変換すると共に、自然画の周波数成分が低周波領域に集中していることを利用して画品質が低下しない範囲内で各係数値を異なる量子化幅で量子化し、情報量の削減を行ってからハフマン符号化を行っている。

【0003】 これに対してHVQ方式は、JPEGと同様に平均値分離型ブロック符号化の一種であるが、ベクトル量子化と直交変換符号化の中間方式である適応的直交変換(AOT: Adaptive Orthogonal Transform)をその圧縮原理としている。ここで、AOTはベクトル量子化のコードブックに相当する基底の巢(ネスト)から必要最少数の非直交基底系を選択し、対象ブロックを所望の許容誤差 $Z$ 以内に近似する方式である。HVQ方式では復号演算を整数型で行えるため、復号が高速である。

またJPEGに特有なモスキート及びブロックノイズ、GIFに特有な擬似輪郭が発生しないため自然画像、人工画像(アニメーション画像、CG画像)を高画質で高圧縮できる。本発明はこのようなHVQ方式における画質の更なる改善及び符号化演算の高速化に関する。

【0004】

【従来の技術】 本件出願人は画像の自己相似性を利用したHVQ方式による画像符号／復号方法を既に提案している(特願平10-189239)。以下その内容を説明する。なお、本明細書を通して記号 $\langle a \rangle$ はベクトル $a$ 又はブロック $a$ 、記号 $\|a\|$ はベクトル $a$ の大きさ(ノルム)、記号 $\langle a \cdot b \rangle$ はベクトル $a$ 、 $b$ の内積を表す。また図や【数】中のベクトルやブロックを太文字で表す。

【0005】 図13は従来の画像符号装置(エンコーダ)のブロック図で、図において、11は原画像データを記憶する原画像メモリ、12は原画像データの各画素ブロック( $4 \times 4$ 画素)につきブロック平均(DC)値を求めるDC値生成部、13は各DC値につき差分予測符号化を行う差分PCM符号部(DPCM)、14は差分PCM符号から各DC値を復号する逆DPCM符号部(IDPCM)、15は復号DC画像を記憶するDC画像メモリ、16はDC画像の一部から所定サイズのDCネストを切り出すDCネスト生成部、17はDCネストを記憶するDCネストメモリである。

【0006】 更に、18は符号対象であるターゲット画像ブロック $\langle R_j \rangle$ から対応する復号DC値 $DC_j$ を分離する減算器、19はDC分離された残差ベクトル $\langle d_j \rangle$ を記憶する残差ベクトルバッファ、20はDCネストからダウンサンプルされた $4 \times 4$ 画素の基底候補ブロック $\langle U_i \rangle$ を記憶する候補ブロックバッファ、21は基底候補ブロック $\langle U_i \rangle$ のブロック平均値 $a_i$ を求める平均器、22は基底候補ブロック $\langle U_i \rangle$ からブロック平均値 $a_i$ を分離する減算器、23は平均値分離された基底候補ベクトル $\langle u_i \rangle$ を記憶する候補ベクトルバッファ、24は、残差ベクトルの二乗ノルム $\|d_j\|^2$ が許容誤差 $Z$ を超える場合に、DCネストを探索して残差ベクトル $\langle d_j \rangle$ を許容誤差 $Z$ 以内に近似するための直交基底系 $\alpha_k \langle u_k' \rangle$  ( $k=1 \sim m$ )を生成する適応的直交変換処理部(AOT)、25は生成された直交基底系 $\alpha_k \langle u_k' \rangle$  ( $k=1 \sim m$ )につき、各対応する非直交基底ベクトル $\langle u_k \rangle$  ( $k=1 \sim m$ )に掛けて等価な非直交基底系 $\beta_k \langle u_k \rangle$  ( $k=1 \sim m$ )を生成するための展開係数 $\beta_k$ を求める係数変換部、26は上記DC値のDPCM符号や非直交基底系 $\beta_k \langle u_k \rangle$ 等の情報を更に圧縮符号化するためのハフマン、ランレングス、固定長符号等による符号部である。

【0007】 DC値生成部12は $4 \times 4$ 画素のブロック平均値を求め、小数点以下を四捨五入(又は切り捨て等)する。DPCM13は、図示しないが、J行、I列

のDC値を $DC_{J,I}$ とする時に、該 $DC_{J,I}$ の予測値 $DC_{J,I}'$ を例えば $DC_{J,I}' = (DC_{J,I-1} + DC_{J-1,I}) / 2$ により求め、その予測誤差 $\Delta DC_{J,I} = DC_{J,I} - DC_{J,I}'$ を量子化係数 $Q(Z)$ により線形量子化して出力する。この量子化係数 $Q(Z)$ は許容誤差 $Z$ と対応付けられており、許容誤差 $Z$ に応じて1～8の範囲で変化する。

【0008】DCネスト生成部16はDC画像から例えば縦39×横71の領域をそのまま切り出(コピー)してDCネストとする。DCネストはコードブックとして使用されるため、交流成分を多く含むものが望ましい。そこで、複数の候補領域につき、各領域内で隣り合うDC値の差分をとってこれらの絶対値等の総和を求め、総和が最大となるような領域を切り出してDCネストとする。

【0009】また基底候補ブロック $\langle U_i \rangle$ のダウンサンプルは、縦横1DC値毎に頂点 $(p_x, p_y) \in [0, 63] \times [0, 31]$ を設定し、かつそのサブサンプル間隔は $(s_x, s_y) \in \{(1, 1), (1, 2), (2, 1), (2, 2)\}$ の計4種類とする。従って、トータルでは $N (= 8192)$ 個の基底候補ブロック $\langle U_i \rangle$ が存在し、これらはAOT24からのインデクスカウンタ $i$ で参照される。以下、従来の適応的直交変換処理部24の動作を説明する。

【0010】図14は従来の適応的直交変換処理のフローチャート、図15で該処理のイメージ図である。図14において、残差ベクトルの二乗ノルム $\| \langle d_j \rangle \|^2 >$

$Z$ であるとの処理に入力する。ステップS121ではレジスタEに残差ベクトルの二乗ノルム $\| \langle d_j \rangle \|^2$ をセットする。また基底数カウンタ $k=1$ に初期化する。ステップS122では最小値保持レジスタE'に大きな値(例えば100000)をセットする。ステップS123では基底候補ブロック $\langle U_i \rangle$ のインデクスカウンタ $i=0$ に初期化する。これはDCネストの開始アドレス $(p_x, p_y) = (0, 0)$ 、サブサンプル間隔 $(s_x, s_y) = (1, 1)$ に対応する。

【0011】ステップS124では基底候補ブロック $\langle U_i \rangle$ からそのブロック平均値 $a_i$ を分離して基底候補ベクトル $\langle u_i \rangle$ を生成する。この演算は整数精度で行われるため、ブロック平均値 $a_i$ に小数点以下の値が発生した場合はこれを四捨五入(又は切り捨て等)する。ステップS125では必要( $k > 1$ )なら基底候補ベクトル $\langle u_i \rangle$ をそれ以前の直交基底ベクトル $\langle u_k \rangle$ に直交化する。

【0012】図15(A)、(B)に直交化処理のイメージ図を示す。図15(A)において、まず第1基底候補ベクトル $\langle u_i \rangle$ はそのままで第1基底ベクトル $\langle u_1' \rangle$ となり得る。次に第2基底候補ベクトル $\langle u_2 \rangle$ は以下の方法により第1基底ベクトル $\langle u_1' \rangle$ に直交化される。即ち、第2基底候補ベクトル $\langle u_2 \rangle$ の第1基底ベクトル $\langle u_1' \rangle$ への射影は(1)式の関係で得られる。

【0013】

【数1】

$$\|u_2\| \cos \theta = \frac{\langle u_1' | u_2 \rangle}{\|u_1'\|} \quad \therefore \langle u_1' | u_2 \rangle = \|u_1'\| \|u_2\| \cos \theta \quad (1)$$

【0014】従って、第2直交ベクトル $\langle u_2' \rangle$ は第2基底候補ベクトル $\langle u_2 \rangle$ から前記射影分のベクトルを引くことで得られる。

【0015】

【数2】

$$u_2' = u_2 - \frac{\langle u_1' | u_2 \rangle}{\|u_1'\|^2} u_1' \quad (2)$$

【0016】図15(B)において、次に第3基底候補ベクトル $\langle u_3 \rangle$ を第1、第2の基底ベクトル

$\langle u_1' \rangle$ 、 $\langle u_2' \rangle$ に直交化する。この図は3次元的に描かれている。まず第3基底候補ベクトル $\langle u_3 \rangle$ を第

1基底ベクトル $\langle u_1' \rangle$ に直交化すると上記同様にして中間の直交ベクトル $\langle u_3' \rangle$ が得られる。

【0017】

【数3】

$$u_3' = u_3 - \frac{\langle u_1' | u_3 \rangle}{\|u_1'\|^2} u_1' \quad (3)$$

【0018】更にこの中間直交ベクトル $\langle u_3' \rangle$ を第2基底ベクトル $\langle u_2' \rangle$ に直交化すると第3基底ベクトル $\langle u_3' \rangle$ が得られる。

【0019】

【数4】

$$\begin{aligned}
\mathbf{u}_3' &= \mathbf{u}_3 - \frac{\langle \mathbf{u}_2', \mathbf{u}_3' \rangle}{\|\mathbf{u}_2'\|^2} \mathbf{u}_2' \\
&= \left( \mathbf{u}_3 - \frac{\langle \mathbf{u}_1', \mathbf{u}_3 \rangle}{\|\mathbf{u}_1'\|^2} \mathbf{u}_1' \right) - \frac{\left\langle \left( \mathbf{u}_3 - \frac{\langle \mathbf{u}_1', \mathbf{u}_3 \rangle}{\|\mathbf{u}_1'\|^2} \mathbf{u}_1' \right), \mathbf{u}_2' \right\rangle}{\|\mathbf{u}_2'\|^2} \mathbf{u}_2' \\
&= \mathbf{u}_3 - \frac{\langle \mathbf{u}_1', \mathbf{u}_3 \rangle}{\|\mathbf{u}_1'\|^2} \mathbf{u}_1' - \frac{\langle \mathbf{u}_2', \mathbf{u}_3 \rangle}{\|\mathbf{u}_2'\|^2} \mathbf{u}_2' \quad (4)
\end{aligned}$$

【0020】図13に戻り、ステップS126では得られた直交ベクトル  $\langle \mathbf{u}_i' \rangle$  を使用し、残差ベクトル  $\langle \mathbf{d}_k \rangle$  (但し、最初は  $\langle \mathbf{d}_j \rangle$ ) との距離を最小とする様なスカラー係数  $\alpha_i$  を求める。

【0021】図15(C)にその処理イメージを示す。図において、ある時点の残差ベクトルを  $\langle \mathbf{d}_k \rangle$  とする時に、これを直交ベクトル  $\langle \mathbf{u}_i' \rangle$  で近似した後の残差ベクトルの二乗ノルム  $e_i = \|\langle \mathbf{d}_k \rangle - \alpha_i \langle \mathbf{u}_i' \rangle\|^2$

$\|^2$  が最小となるのは、図より明らかなように、直交ベクトル  $\langle \mathbf{u}_i' \rangle$  にスカラー係数  $\alpha_i$  を掛けたものと、残差ベクトル  $\langle \mathbf{d}_k \rangle - \alpha_i \langle \mathbf{u}_i' \rangle$  とが直交する時 (内積=0) である。従って、スカラー係数  $\alpha_i$  は (5) 式の関係により求まる。

【0022】

【数5】

$$\begin{aligned}
\langle \alpha_i \mathbf{u}_i', \mathbf{d}_k - \alpha_i \mathbf{u}_i' \rangle &= 0 & \alpha_i \langle \mathbf{u}_i', \mathbf{d}_k \rangle - \alpha_i^2 \langle \mathbf{u}_i', \mathbf{u}_i' \rangle &= 0 & (5-1) \\
\alpha_i &= \frac{\langle \mathbf{d}_k, \mathbf{u}_i' \rangle}{\|\mathbf{u}_i'\|^2} & (5-2)
\end{aligned}$$

【0023】なお、図には残差ベクトル  $\langle \mathbf{d}_k \rangle$  (但し、 $k=0$ ) を他の第1基底候補ベクトル  $\langle \mathbf{u}_{j'} \rangle$  で近似した場合が描かれている。第1基底候補ベクトル  $\langle \mathbf{u}_{j'} \rangle$  は任意方向をとり得るから、図示のようなイメージとなる。

【0024】図14に戻り、ステップS127では残差

ベクトル  $\langle \mathbf{d}_k \rangle$  を基底候補ベクトル  $\alpha_i \langle \mathbf{u}_i' \rangle$  で近似した後の誤差ベクトルの二乗ノルム  $e_i$  を求める。この演算は (6) 式により得られる。

【0025】

【数6】

$$\begin{aligned}
e_i &= \|\mathbf{d}_k - \alpha_i \mathbf{u}_i'\|^2 \\
&= \|\mathbf{d}_k\|^2 - 2\alpha_i \langle \mathbf{d}_k, \mathbf{u}_i' \rangle + \alpha_i^2 \|\mathbf{u}_i'\|^2 \\
&= \|\mathbf{d}_k\|^2 - 2 \frac{\langle \mathbf{d}_k, \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} + \frac{\langle \mathbf{d}_k, \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^4} \|\mathbf{u}_i'\|^2 \\
&= \|\mathbf{d}_k\|^2 - \frac{\langle \mathbf{d}_k, \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} \\
&= E - \frac{\langle \mathbf{d}_k, \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} \quad (6)
\end{aligned}$$

【0026】ステップS128では  $e_i < E'$  か否かを判別する。  $e_i < E'$  の場合はステップS129で  $E'$  の内容を  $e_i$  で更新する。またその時の  $\alpha_i$ ,  $\langle \mathbf{u}_i' \rangle$ ,  $\langle \mathbf{u}_i \rangle$  等に係る情報を配列  $[\alpha_k]$ ,  $[\mathbf{u}_k']$ ,  $[\mathbf{u}_k]$  に保持する。また  $e_i < E'$  でない場合は上記ステップS129の処理をスキップする。

【0027】ステップS130ではカウンタ  $i$  に+1し、更にステップS131では  $i \geq N$  ( $=8192$ ) か否かを判別する。  $i \geq N$  でない場合はステップS124に戻り、次の基底候補ベクトル  $\langle \mathbf{u}_i \rangle$  につき上記同様

の処理を行う。以下同様にして進み、やがて、ステップS131の判別で  $i \geq N$  になるとこの段階における全基底候補ベクトル  $\langle \mathbf{u}_i \rangle$  が試されたことになる。この時、レジスタ  $E'$  は最小の二乗ノルム  $e_i$  を保持している。

【0028】ステップS132では  $E' \leq Z$  か否かを判別し、  $E' \leq Z$  でない場合はステップS133で  $E = E'$  とする。即ち、残差ベクトルの二乗ノルムを更新する。ステップS134では  $k$  に+1し、ステップS122に戻る。また  $E' \leq Z$  の場合はこの処理を抜ける。こ

うして、最初の残差ベクトル  $\langle d_j \rangle$  との差を許容誤差  $Z$  以下に近似するための直交基底系  $\alpha_k \langle u_k' \rangle$  ( $k = 1 \sim m$ ) が得られる。

【0029】

【発明が解決しようとする課題】しかし、上記従来方式では基底候補ブロック  $\langle U_i \rangle$  のブロック平均値  $a_i$  につきその小数点以下を四捨五入（又は切り捨て等）していたため、画質の改善が頭打ちとなる不都合があった。これを図16に従って説明する。

【0030】図16(a)は基底候補ブロック  $\langle U_i \rangle$  のある行の画素値を列(x)方向に見た場合を示している。実際は16画素分のブロック平均値であるが、ここでは説明の簡単のため4画素で説明する。図16(a)において、各画素値は「5, 2, 4, 3」からなりそのブロック平均値  $a_i = 3.5$  である。今、例えばこの小数点以下を切り捨てると、図16(b)に示す如く、基底候補ベクトル  $\langle u_i \rangle$  のブロック平均値  $a_i = 0.5$  となる。図16(c)において、復号ブロックのDC値  $DC_j$  に基底ベクトル  $\beta_k \langle u_k \rangle$  を加算すると、復号画像のターゲットブロック  $\langle R_j \rangle$  にはDC成分 ( $a_i = 0.5$ ) が重畳されてしまう。しかも、基底数が複数の場合は、このようなDC成分は  $0 < a_i < 1$  の範囲の様々な値で  $DC_j$  に重畳される結果、復号画像ではブロック毎に一種の雑音为重畳された形となり、このため画質の改善が図れなかった。以上のことは小数点以下を四捨五入又は切り上げる場合も同様である。

【0031】また、従来のAOT処理では各基底候補ベクトル  $\langle u_i \rangle$  を一々前の基底ベクトル  $\langle u_k' \rangle$  に直交化していたため、AOT処理に多大の演算と時間を要していた。

【0032】本発明は上記従来技術の問題点を鑑み成されたもので、その目的とする所は、より高画質かつ高速の符号／復号が得られる画像符号／復号方法及びその装置並びにそのプログラムを記録した記録媒体を提供することにある。

【0033】

【課題を解決するための手段】上記の課題は例えば図1の構成により解決される。即ち、本発明(1)の画像符号方法は、画像データをB画素毎にブロック分割して各ブロック平均値からなるDC画像を生成し、その一部をDCネストとすると共に、符号対象の画素ブロック  $\langle R_j \rangle$  からそのDC値  $DC_j$  を分離した残差ベクトル  $\langle d_j \rangle$  の大きさが許容値  $Z$  を超える場合は、該残差ベクトル  $\langle d_j \rangle$  を近似するための1又は2以上の直交基底 ( $\alpha_k \langle v_k \rangle$  等) をDCネストを使用した適応的直交変換 (AOT) により求める画像符号方法において、DCネストから基底候補ブロック  $\langle U_i \rangle$  をダウンサンプルしてそのブロック平均値  $a_i$  を求める際の各サンプルDC画素の下位  $n$  ( $n = \log_2 B$ ) ビットが0にされているものである。従って、ブロック平均値  $a_i$  に小数点以下

の端数は生ぜず、整数精度のブロック平均値  $a_i$  が高速に得られる。

【0034】好ましくは本発明(2)においては、上記本発明(1)において、DC画像からDCネストを生成する際に各DC画素の下位  $n$  ビットを0に（マスク）する。従って、1回の処理でその下位  $n$  ビットを0にされたDCネストが効率よく得られる。

【0035】また好ましくは本発明(3)においては、上記本発明(1)又は(2)において、下位  $n$  ビットが0にされている基底候補ブロック  $\langle U_i \rangle$  からそのブロック平均値  $a_i$  を分離して残差ベクトル  $\langle d_j \rangle$  を近似するための基底候補ベクトル  $\langle u_i \rangle$  を生成する。

【0036】本発明(3)によれば、このような基底候補ベクトル  $\langle u_i \rangle$  はその全要素の和（ブロック平均値）が常に0であり、DC成分が完全に分離されている。従って、復号側でこのような基底ベクトル  $\langle u_k \rangle$  を幾つ重ねても不要なDC成分（雑音）は生じない。そして、これにより本HVQ方式の画質が大幅に改善された。

【0037】また好ましくは本発明(4)においては、上記本発明(3)において、基底候補ベクトル  $\langle u_i \rangle$  の任意要素（例えば  $u_{16}$ ）を残りの要素の一次結合で置き換えると共に、該基底候補ベクトル  $\langle u_i \rangle$  と任意他のベクトル  $\langle w \rangle$  との内積を、

$$\langle w \cdot u_i \rangle = (w_1 - w_{16}) u_1 + (w_2 - w_{16}) u_2 + \dots + (w_{15} - w_{16}) u_{15}$$

の積和演算により求める。

【0038】本発明(4)においては、上記基底候補ベクトル  $\langle u_i \rangle$  の全要素の和が常に0であることにより、その任意要素（例えば  $u_{16}$ ）を残りの要素の一次結合で表せる。従って、任意他のベクトル  $\langle w \rangle$  との内積演算  $\langle w \cdot u_i \rangle$  は上式のような積和演算に展開でき、こうして面倒な積和演算の回数を1回分省略できる。HVQ方式による画像符号処理ではベクトルの内積演算が大量に行われるため、各1回の省略は全体としての符号処理の高速化に大きく貢献する。

【0039】また好ましくは本発明(5)においては、上記本発明(3)又は(4)において、残差ベクトル  $\langle d \rangle$ 、基底候補ベクトル  $\langle u_i \rangle$  とするとき第1基底の探索は、

$$h_1 = \langle d \cdot u_1 \rangle^2 / \|u_1\|^2$$

を最大とするものを条件に探索する。

【0040】本発明(5)によれば、図14の従来のステップS126, S127で行っていたような残差ベクトル  $\langle d \rangle$  との差の二乗ノルム  $\| \langle d \rangle - \alpha_1 \langle u_1 \rangle \|^2$  を最小とするような条件を上記簡単な演算及び条件で探索できる。従って、AOT処理を高速化できる。

【0041】また好ましくは本発明(6)においては、上記本発明(3)又は(4)において、残差ベクトル  $\langle d \rangle$ 、第1基底に対応する基底候補ベクトル

$\langle u_1 \rangle$ 、第2基底を探索する基底候補ベクトル  $\langle u_1 \rangle$  とするときに第2基底の探索は、

$$h_1 = \{ \langle d \cdot u_1 \rangle - ( \langle d \cdot u_1 \rangle \langle u_1 \cdot u_1 \rangle ) / \| u_1 \|^2 \}^2 / \{ \| u_1 \|^2 - ( \langle u_1 \cdot u_1 \rangle ) / \| u_1 \|^2 \}$$

を最大とするものを条件に探索する。

【0042】本発明(6)によれば、上記本発明(5)の効果に加え、上式分子の  $\langle d \cdot u_1 \rangle$ 、 $\| u_1 \|^2$  及び分母の  $\| u_1 \|^2$ 、 $\| u_1 \|^2$  については既に第1基底探索で行った演算結果を利用できるため、AOT処理を更に効率化、高速化できる。

【0043】また好ましくは本発明(7)においては、上記本発明(3)又は(4)において、残差ベクトル  $\langle d \rangle$ 、第1正規基底ベクトル  $\langle v_1 \rangle$ 、第2正規直交基底ベクトル  $\langle v_2 \rangle$ 、第3基底を探索する基底候補ベクトル  $\langle u_1 \rangle$  とするときに第3基底の探索は、

$$h_1 = ( \langle d \cdot u_1 \rangle - ( \langle d \cdot v_1 \rangle \langle v_1 \cdot u_1 \rangle - \langle d \cdot v_2 \rangle \langle v_2 \cdot u_1 \rangle )^2 / \{ \| u_1 \|^2 - \langle v_1 \cdot u_1 \rangle^2 - \langle v_2 \cdot u_1 \rangle^2 \} )^2$$

を最大とするものを条件に探索する。

【0044】本発明(7)によれば、上記本発明(5)、(6)の効果に加え、上式分子の  $( \langle d \cdot u_1 \rangle - ( \langle d \cdot v_1 \rangle \langle v_1 \cdot u_1 \rangle )$  及び分母の  $( \| u_1 \|^2 - \langle v_1 \cdot u_1 \rangle^2 )$  については既に第1、第2の基底探索で行った演算結果を利用できるため、こうしてAOT処理を更に効率化、高速化できる。

【0045】また好ましくは本発明(8)においては、上記本発明(6)又は(7)において、探索条件にマッチした基底候補ベクトル  $\langle u_1 \rangle$  をそれ以前の1又は2以上の正規直交基底に正規直交化する。

【0046】即ち、このような正規直交化は各段階の探索終了により基底に採用された各基底候補ベクトル  $\langle u_i \rangle$  につき夫々1回だけ行えばよく、こうしてAOT処理を更に効率化、高速化できる。

【0047】また本発明(9)の画像符号方法は、上記前提となる画像符号方法において、求められた基底系を  $\beta_k \langle u_k \rangle$  ( $k=1 \sim m$ ) とするときに、各スカラー展開係数  $\beta_1 \sim \beta_m$  のノルムをその大きさに並べ替え、0を含む隣接ノルム間の各差分を求め、得られた各差分につきハフマン符号を適用するものである。

【0048】一般にスカラー展開係数  $\beta_1 \sim \beta_m$  のノルムは様々な値をとり得るが、これらを大きさに並べて0を含む隣接ノルム間の各差分をとると、各差分の大きさは互いに近似(又は同一)となる場合が少なくない。そこで、これらの差分値にハフマン符号を適用することで更なる符号圧縮が可能となる。

【0049】また本発明(10)の画像符号方法は、上記前提となる画像符号方法において、求めた基底数が所定以上の場合は基底系の符号化に代えて、符号対象ブロックの画像データ  $\langle R_j \rangle$  そのものを符号化するもので

ある。従って、復号画質の改善が図れる。また實際上このような状況は極めて少ないので符号圧縮率に与える影響は極めて少ない。

【0050】また上記の課題は例えば図10の構成により解決される。即ち、本発明(11)の画像復号方法は、HVQ方式に係る符号データからB画素毎の各ブロック平均値に相当するDC画像を再生し、その一部をDCネストとすると共に、ターゲットブロックのDC値DC<sub>Cj</sub>に対し、前記符号データに基づきDCネストから選択生成した1又は2以上の基底ベクトル  $\beta_k \langle u_k \rangle$  を合成してターゲットブロックの画像データ  $\langle R_j \rangle$  を再生する画像復号方法において、DCネストから選択ブロック  $\langle U_k \rangle$  をダウンサンプルしてそのブロック平均値を求める際の各サンプルDC画素の下位n( $n=\log_2 B$ )ビットが0にされているものである。従って、ブロック平均値に小数点以下の端数は生ぜず、整数精度のブロック平均値が高速に得られる。

【0051】また本発明(12)の画像復号方法は、上記前提となる画像復号方法において、復号された基底系が  $\beta_k \langle u_k \rangle$  ( $k=1 \sim m$ ) に係る情報であるときに、DCネストから読み出した各選択ブロック  $\langle U_k \rangle$  につき各DC画素の下位n( $n=\log_2 B$ )ビットが0にされていると共に、まず  $\beta_k \langle U_k \rangle$  ( $k=1 \sim m$ ) の積和演算を行い、その演算結果をブロック画素数Bで除算するものである。

【0052】本発明(12)においては、各選択ブロック  $\langle U_k \rangle$  の下位nビットが0にされていることにより、これらを予め累積加算しても、その加算結果はブロックサイズB(例えば16)の整数倍となる。なお、展開係数  $\beta_k$  は整数精度とする。従って、最後にこの累積加算結果をブロック画素数Bで除算すれば1回の除算でブロック平均値  $A_j$  が効率よく求まる。従って、基底ベクトル  $\beta_k \langle u_k \rangle$  ( $k=1 \sim m$ ) を重ね合わせる演算を効率よく行える。

【0053】好ましくは本発明(13)においては、上記本発明(11)又は(12)において、DC画像からDCネストを生成する際に各DC画素の下位nビットを0にする。従って、処理効率が良い。

【0054】また本発明(14)の画像符号装置は、画像データをB画素毎にブロック分割して各ブロック平均値からなるDC画像を生成し、その一部をDCネストとすると共に、符号対象の画素ブロック  $\langle R_j \rangle$  からそのDC値DC<sub>Cj</sub>を分離した残差ベクトル  $\langle d_j \rangle$  の大きさが許容値Zを超える場合は、該残差ベクトル  $\langle d_j \rangle$  を近似するための1又は2以上の直交基底 ( $\alpha_k \langle v_k \rangle$  等) をDCネストを使用した適応的直交変換(AOT)により求める画像符号装置において、各DCネスト画素の下位n( $n=\log_2 B$ )ビットが0にされているDCネストを記憶するメモリ17を備えるものである。

【0055】また本発明(15)の画像復号装置は、H



VQ方式に係る符号データからB画素毎の各ブロック平均値に相当するDC画像を再生し、その一部をDCネストとすると共に、ターゲットブロックのDC値 $DC_j$ に対し、前記符号データに基づきDCネストから選択生成した1又は2以上の基底ベクトル $\beta_k \langle u_k \rangle$ を合成してターゲットブロックの画像データ $\langle R_j \rangle$ を再生する画像復号装置において、各DCネスト画素の下位 $n$  ( $n = \log_2 B$ ) ビットが0にされているDCネストを記憶するメモリ49を備えるものである。

【0056】また本発明(16)の記録媒体は、上記本発明(1) 1乃至(13)の何れか1つに記載の処理をコンピュータに実行させるためのプログラムを記録したコンピュータ読取り可能な記録媒体である。

【0057】

【発明の実施の形態】以下、添付図面に従って本発明に好適なる実施の形態を詳細に説明する。なお、全図を通して同一符号は同一又は相当部分を示すものとする。

【0058】図2は実施の形態による画像符号装置のブロック図で、図において、31は復号DC画像から本発明によるDCネストを生成するDCネスト生成部、17は生成されたDCネストを記憶するDCネストメモリ、32はAOT処理を効率よくかつ高速に行う適応的直交変換処理部(AOT)、33は係数変換部、34は展開係数 $\beta_k$ の更に高圧縮を可能とする符号部である。その他の構成については上記図13で述べたものと同様でよい。なお、上記各部の特徴は以下の動作説明によって明らかとなる。

【0059】図3は実施の形態による画像符号(メイン)処理のフローチャートである。ステップS1では原画像メモリ11に原画像データを読み込む。例えばRGB系の対象画像をYUV系に変換して読み込む。Yは輝度データ、U、Vは色差データに相当し、U、Vは横2画素の輝度平均を用いてダウンサンプリングされる。一例の輝度データYは縦960×横1280画素からなり、画素毎に例えば8ビットが割り付けられている。なお、以下は輝度データYの処理を中心に述べるが、U、Vについても同様に処理できる。

【0060】ステップS2では全画像データにつき4×4画素毎のブロック平均(DC)値を求める。このとき小数点以下は例えば四捨五入される。ステップS3では全DC値を公知の2次元DPCM法等により符号化して出力する。ステップS4では全DPCM出力をIDPCM復号してDC画像を再生し、DC画像メモリ15に格納する。これは符号側/復号側のAOT処理条件を同一にするためである。ステップS5ではDCネスト生成部31がDC画像からDCネストを生成し、DCネストメモリ17に格納する。なお、DCネストを切り出す領域の選択等は従来と同様でよい。

【0061】図7にDCネストの生成イメージを示す。図7(a)において、本実施の形態ではDC画像メモリ

15から切り出した各DC画素 $DC_j$ の下位4ビットをマスク(=0)してこれをDCネストメモリ17のネスト画素 $N_j$ に記憶する。下位4ビットは $2^4=B$  ( $B$ =ブロックサイズ16)又は $4=\log_2 B$ の関係にある。下位4ビットをマスクした結果、基底候補ブロック $\langle U_i \rangle$ の総和は常に16の整数倍となり、よってこれを1/16したブロック平均値 $a_i$ は常に整数となる。従って、基底候補ブロック $\langle U_i \rangle$ からブロック平均値 $a_i$ を分離した基底候補ベクトル $\langle u_i \rangle$ のブロック平均値は常に0となる。

【0062】図7(a)、(b)に具体的な数値例をグラフで示す。但し、ここでは説明の簡単のため4画素分の平均をとっている。図7(c)において、復号ブロック $\langle R_j \rangle$ のDC値 $DC_j$ に複数の基底ベクトル $\beta_k \langle u_k \rangle$ を累積加算しても、各基底ベクトル $\beta_k \langle u_k \rangle$ のブロック平均値は常に0であるため、従来のような雑音は重畳されない。これにより画質の大幅な改善が図れた。

【0063】図8(a)に図7の数値例を表で示す。DC画素A~Dの合計SUM=251であり、その平均値 $AV=251/4=62.75$ (非整数)である。これらのDC画素A~Dをネスト画素A~Dに転送する際に下位4ビットをマスクする。これによりネスト画素A~Dの合計SUM=224となり、その平均値 $AV=224/4=56$ (整数)となる。更に、ネスト画素A~Dからその平均値 $AV=56$ を分離した基底候補ベクトル $\langle u_i \rangle$ の各要素 $a \sim d$ は「24, -24, 8, -8」となり、これらの総和sum=0(完全平均値分離)となっている。

【0064】図8(b)は図8(a)と同じ数値例を示している。但し、DC画素A~Dをそのままネスト画素A~Dにコピーし、ネスト画素A~Dの総和SUMから下位4ビットをマスク(=0)する点で異なっている。この方法でも総和SUMは16の倍数になるから、ブロック平均値 $AV=60$ (整数)となる。しかしこの方法によると、ネスト画素A~Dからその平均値 $AV=60$ を分離した基底候補ベクトル $\langle u_i \rangle$ の各要素 $a \sim d$ は「33, -25, 13, -10」となり、必ずしもその総和sum=0(完全平均値分離)とはならない。

【0065】なお、図8(b)に示す如くDC画像の一部をそのままDCネストにコピーしておき、該DCネストから基底候補ブロック $\langle U_i \rangle$ をダウンサンプルする時に各画素から下位4ビットをマスク(=0)しても良い。

【0066】図3に戻り、ステップS6では原画像メモリ11及びDC画像メモリ15に対する各インデックスカウンタj、Jを共に0に初期化する。但し、jは符号対象のターゲットブロック $\langle R_j \rangle$ のインデックスカウンタ、JはDC画素のインデックスカウンタを夫々表す。ステップS7ではターゲットブロック $\langle R_j \rangle$ から対応す

る復号DC値DC<sub>j</sub>を分離して残差ベクトル〈d<sub>j</sub>〉を求める。ステップS 8では残差ベクトルの二乗ノルム $\|d_j\|^2$ が許容誤差Zより大きいかな否かを判別する。 $\|d_j\|^2 > Z$ でない場合はステップS 17で基底数「0」を符号出力する。この場合のターゲットブロック〈R<sub>j</sub>〉は後述の交流成分予測法により復号される。また $\|d_j\|^2 > Z$ の場合はステップS 9で後述の適応的直交変換処理を行う。

【0067】ステップS 10では適応的直交変換で生成された基底数k > 4かな否かを判別する。因みに、実測ではほとんどの場合にk = 1～3程度の統計結果が得られている。そこで、k > 4の場合はステップS 18で基底数「5」を符号出力し、かつターゲットブロック

〈R<sub>j</sub>〉の各画素値を符号出力する。またk > 4でない場合はステップS 11で後述の展開係数 $\beta_k$ への変換を行う。ステップS 12では基底数「m」、展開係数 $\beta_k$ 及び非直交基底ベクトル〈u<sub>i</sub>〉のインデクス情報iを夫々符号出力する。

【0068】ステップS 13ではカウンタj、Jに夫々+1する。但し、カウンタjに対する+1は1画素ブロック分の更新を意味する。ステップS 14ではj ≥ M (=全画像ブロック数)かな否かを判別する。j ≥ Mでな

い場合はステップS 7に戻り、次のターゲットブロック〈R<sub>j</sub>〉につき上記同様の符号処理を行う。以下同様にして進み、やがて、ステップS 14の判別でj ≥ Mになると、ステップS 15ではハフマン等による符号化を行う。この符号化については後述する。こうして1画像分の符号処理を終了する。

【0069】図4～図6は実施の形態による適応的直交変換処理のフローチャート(1)～(3)で、必要最少数の直交基底系 $\alpha_k$ 〈v<sub>k</sub>〉(k = 1～m)を効率よくかつ高速に求め得る場合を示している。なお、以下の説明では上記ステップS 7で求められた最初の残差ベクトル〈d<sub>j</sub>〉を〈d〉で表し、その後更新される残差ベクトルを〈d<sub>k</sub>〉(k = 1～m)で表す。

【0070】図4は第1基底の探索処理を示している。この処理の説明前に、該処理を高速に行うために行った計算上の工夫を説明する。即ち、通常なら第1基底は残差ベクトル〈d〉との差の二乗ノルムe<sub>1</sub>を最小とする基底候補ベクトル〈u<sub>1</sub>〉として求められるが、この関係式を更に展開すると(7)式が得られる。

【0071】

【数7】

$$\begin{aligned} e_1 &= \left\| d - \frac{\langle d \cdot u_1 \rangle}{\|u_1\|^2} u_1 \right\|^2 \\ &= \|d\|^2 - 2 \frac{\langle d \cdot u_1 \rangle^2}{\|u_1\|^2} + \frac{\langle d \cdot u_1 \rangle^2}{\|u_1\|^4} \|u_1\|^2 \\ &= \|d\|^2 - \frac{\langle d \cdot u_1 \rangle^2}{\|u_1\|^2} \end{aligned} \quad \text{但し、} \quad u_1 = u_1' \quad (7)$$

【0072】ところで、(7)式右辺第1項の $\|d\|^2 > 0$ は基底候補によらず一定であるから、同右辺第2項を最大にする〈u<sub>1</sub>〉が第1基底となり得る。そこで、こ

の右辺第2項をh<sub>1</sub>とおく。

【0073】

【数8】

$$h_1 = \frac{\langle d \cdot u_1 \rangle^2}{\|u_1\|^2} \quad (8)$$

【0074】以下は、h<sub>1</sub>を最大とするような第1基底 $\alpha_k$ 〈v<sub>k</sub>〉を探索・決定する処理である。ステップS 21では後述の内積演算〈d・u<sub>1</sub>〉の前処理として〈d〉の第16成分を残りの各成分の値から差し引いた15次元ベクトル〈d'〉を求める。ステップS 22ではi = 0～(N-1)につきh<sub>1</sub>分子の内積〈d'・u<sub>i</sub>〉を求め、これらを配列[P<sub>i</sub>] {i = 0～(N-1)}に格納する。

【0075】この内積演算を具体的に言うと、〈u<sub>i</sub>〉は本来16次元ベクトルであるが、本実施の形態ではそのブロック平均値(全要素の和) = 0により、その第16成分u<sub>16</sub>は残りの15成分の一次結合で表せる。

【0076】

【数9】

$$\begin{aligned} u_i &= [u_1, u_2, u_3, \dots, u_{16}] \\ u_1 + u_2 + \dots + u_{16} &= 0 \\ u_{16} &= -(u_1 + u_2 + \dots + u_{15}) \end{aligned} \quad (9)$$

【0077】従って、h<sub>1</sub>分子の内積〈d・u<sub>i</sub>〉をこれと等価な〈d'・u<sub>i</sub>〉により求め、こうして積和演算を1回分(全iでは8192回分)省略できる。

【0078】

【数10】

$$\begin{aligned}\langle d \bullet u_i \rangle &= d_1 u_1 + d_2 u_2 + \dots + d_{15} u_{15} - d_{16} (u_1 + u_2 + \dots + u_{15}) \\ &= (d_1 - d_{16}) u_1 + (d_2 - d_{16}) u_2 + \dots + (d_{15} - d_{16}) u_{15} \\ &= \langle d' \bullet u_i \rangle\end{aligned}\quad (10-1)$$

$$d' = [(d_1 - d_{16}), (d_2 - d_{16}), \dots, (d_{15} - d_{16})] \quad (10-2)$$

【0079】ステップS23では $i=0 \sim (N-1)$ につき $h_i$ 分母の二乗ノルム $\|u_i\|^2$ を求め、これらを配列 $[L_i]$  ( $i=0 \sim (N-1)$ )に格納する。

【0080】

【数11】

$$\|u_i\|^2 = u_1^2 + u_2^2 + \dots + u_{16}^2 \quad (11)$$

【0081】配列 $[L_i]$ は一度求めておけば後に結果を利用できる。ステップS24では $h_i$ の最大値を保持するレジスタ $E=0$ 、基底候補ベクトル $\langle u_i \rangle$ のインデックスカウンタ $i=0$ 、基底数カウンタ $k=1$ に夫々初期化する。

【0082】ステップS25では $h_i = P_1^2 / L_i$ を求める。ステップS26では $h_i > E$ か否かを判別する。 $h_i > E$ の場合はステップS27で $E$ を $h_i$ で更新し、かつその時の $i$ を配列 $[I_k]$  ( $k=1$ )に保持する。また $h_i > E$ でない場合は上記ステップS27の処理をスキップする。

【0083】ステップS28では $i$ に $+1$ し、更にステップS29では $i \geq N$  (全候補数)か否かを判別する。 $i \geq N$ でない場合はステップS25に戻り、次の $h_i$ につき上記同様の最大値探索処理を行う。以下、同様にして進み、やがて $i \geq N$ になると全ネストブロックの探索終了である。この時、上記配列 $[I_k]$ には $h_i$ を最大とするような第1基底ベクトル $\langle u_i \rangle$ のインデックス値 $i$ が保持されている。

【0084】ステップS30では第1基底ベクトル $\langle u$

$\rangle$ を正規化して正規化基底ベクトル $\langle v_1 \rangle$ となし、これを配列 $[V_k]$  ( $k=1$ )に格納する。またスカラー係数 $\alpha_1$  ( $\langle d \rangle$ の $\langle v_1 \rangle$ への射影)を求め、これを配列 $[A_k]$  ( $k=1$ )に格納する。

【0085】ステップS31では残差ベクトル $\langle d \rangle$ を第1基底で近似後の残差ベクトル $\langle d_1 \rangle = \langle d \rangle - \alpha_1 \langle v_1 \rangle$ により更新する。ステップS32では新たな残差ベクトルの二乗ノルム $e = \|\langle d_1 \rangle\|^2$ を求め、更にステップS33では $e \leq Z$ か否かを判別する。 $e \leq Z$ の場合はこの段階でAOT処理を終了し、また $e \leq Z$ でない場合は次いで第2基底の探索処理を行う。

【0086】図5は第2基底の探索処理を示している。この処理の説明前に該処理を効率よく行うために行った計算上の工夫を説明する。即ち、通常なら第2基底は残差ベクトル $\langle d_1 \rangle$ との差の二乗ノルム $e_1$ を最小とする直交ベクトル $\langle u_1' \rangle$ として求められるが、この関係式を更に展開すると(12)式が得られる。

【0087】

【数12】

$$\begin{aligned}e_1 &= \left\| d_1 - \frac{\langle d_1 \bullet u_1' \rangle}{\|u_1'\|^2} u_1' \right\|^2 \\ &= \|d_1\|^2 - 2 \frac{\langle d_1 \bullet u_1' \rangle^2}{\|u_1'\|^2} + \frac{\langle d_1 \bullet u_1' \rangle^2}{\|u_1'\|^4} \|u_1'\|^2 \\ &= \|d_1\|^2 - \frac{\langle d_1 \bullet u_1' \rangle^2}{\|u_1'\|^2}\end{aligned}\quad (12)$$

【0088】ここで、直交ベクトル $\langle u_1' \rangle$ は第2基底の候補ベクトル $\langle u_i \rangle$ を第1正規化基底ベクトル $\langle v_1 \rangle$ に直交化したものである。

【0089】

【数13】

$$u_1' = u_i - \frac{\langle u_i \bullet v_1 \rangle}{\|v_1\|^2} v_1 = u_i - \langle u_i \bullet v_1 \rangle v_1 \quad (13)$$

【0090】同様にして、上記(12)式右辺第1項の $\|d_1\|^2 > 0$ は基底候補によらず一定であるから、同右辺第2項を最大にするような直交ベクトル $\langle u_1' \rangle$ が第2基底となり得る。この右辺第2項を $h_1$ とおく。

【0091】

【数14】

$$h_i = \frac{\langle \mathbf{d}_1 \cdot \mathbf{u}_i \rangle^2}{\|\mathbf{u}_i\|^2} \quad (14)$$

【0092】  $h_i$  は、このまま求めても良いが、上記図4の演算結果を効率よく利用するために(14)式の分

母を変形する。即ち、まず  $h_i$  分子の直交ベクトル  $\langle \mathbf{u}_i' \rangle$  を基底候補ベクトル  $\langle \mathbf{u}_i \rangle$  で表すと、 $h_i$  分子は(15)式で表せる。

【0093】

【数15】

$$\begin{aligned} \langle \mathbf{d}_1 \cdot \mathbf{u}_i' \rangle^2 &= \langle \mathbf{d}_1 \cdot (\mathbf{u}_i - \langle \mathbf{u}_i \cdot \mathbf{v}_1 \rangle \mathbf{v}_1) \rangle^2 \\ &= \langle (\mathbf{d}_1 \cdot \mathbf{u}_i) - \mathbf{d}_1 \cdot \langle \mathbf{u}_i \cdot \mathbf{v}_1 \rangle \mathbf{v}_1 \rangle^2 \\ &= \langle \mathbf{d}_1 \cdot \mathbf{u}_i \rangle^2 \quad \because \langle \mathbf{d}_1 \cdot \mathbf{v}_1 \rangle = 0 \end{aligned} \quad (15)$$

【0094】更に残差ベクトル  $\langle \mathbf{d}_1 \rangle$  を最初の残差ベクトル  $\langle \mathbf{d} \rangle$  で表すと、 $h_i$  分子は(16)式で表せる。

【0095】

【数16】

$$\begin{aligned} \langle \mathbf{d}_1 \cdot \mathbf{u}_i \rangle^2 &= \langle (\mathbf{d} - \langle \mathbf{d} \cdot \mathbf{v}_1 \rangle \mathbf{v}_1) \cdot \mathbf{u}_i \rangle^2 \\ &= \langle (\mathbf{d} \cdot \mathbf{u}_i) - \langle \mathbf{d} \cdot \mathbf{v}_1 \rangle \langle \mathbf{v}_1 \cdot \mathbf{u}_i \rangle \rangle^2 \\ &= \left( \langle \mathbf{d} \cdot \mathbf{u}_i \rangle - \frac{\langle \mathbf{d} \cdot \mathbf{u}_1 \rangle \langle \mathbf{u}_1 \cdot \mathbf{u}_i \rangle}{\|\mathbf{u}_1\|} \right)^2 \end{aligned} \quad (16)$$

【0096】従って、 $h_i$  分子の演算には第1基底探索で得られた演算結果  $\langle \mathbf{d} \cdot \mathbf{u}_1 \rangle$ 、 $\|\mathbf{u}_1\|$  を利用できる。また、同様にして  $h_i$  分母を変形すると(17)式

【0097】

【数17】

$$\begin{aligned} \|\mathbf{u}_i\|^2 &= \|\mathbf{u}_i - \langle \mathbf{u}_i \cdot \mathbf{v}_1 \rangle \mathbf{v}_1\|^2 \\ &= \|\mathbf{u}_i\|^2 - 2\langle \mathbf{u}_i \cdot \mathbf{v}_1 \rangle^2 + \langle \mathbf{u}_i \cdot \mathbf{v}_1 \rangle^2 \|\mathbf{v}_1\|^2 \\ &= \|\mathbf{u}_i\|^2 - \langle \mathbf{u}_i \cdot \mathbf{v}_1 \rangle^2 \\ &= \|\mathbf{u}_i\|^2 - \left( \frac{\langle \mathbf{u}_1 \cdot \mathbf{u}_i \rangle}{\|\mathbf{u}_1\|} \right)^2 \end{aligned} \quad (17)$$

【0098】従って、 $h_i$  分母の演算には第1基底探索で得られた演算結果  $\|\mathbf{u}_1\|^2$ 、 $\|\mathbf{u}_1\|$  を利用できる。以上を(14)式の  $h_i$  に代入すると、 $h_i$  は(18-

1) 式となり、最終的に(18-2)式で表せる。

【0099】

【数18】

$$h_i = \frac{\left( \langle \mathbf{d} \cdot \mathbf{u}_i \rangle - \frac{\langle \mathbf{d} \cdot \mathbf{u}_1 \rangle \langle \mathbf{u}_1 \cdot \mathbf{u}_i \rangle}{\|\mathbf{u}_1\|} \right)^2}{\|\mathbf{u}_i\|^2 - \left( \frac{\langle \mathbf{u}_1 \cdot \mathbf{u}_i \rangle}{\|\mathbf{u}_1\|} \right)^2} \quad (18-1)$$

$$= \frac{\left( P_i - \frac{P_1}{\sqrt{L_k}} \frac{\langle \mathbf{u}_k \cdot \mathbf{u}_i \rangle}{\sqrt{L_k}} \right)^2}{L_i - \left( \frac{\langle \mathbf{u}_k \cdot \mathbf{u}_i \rangle}{\sqrt{L_k}} \right)^2} \quad (18-2)$$

【0100】ここで、 $P_i = \langle \mathbf{d} \cdot \mathbf{u}_i \rangle$ 、 $L_i = \|\mathbf{u}_i\|^2$  は配列  $[P_i]$ 、 $[L_i]$  の演算結果を夫々利用できる。かつ  $P_k = P_1 = \langle \mathbf{d} \cdot \mathbf{u}_1 \rangle$ 、 $\sqrt{L_k} = \sqrt{L_1} = \|\mathbf{u}_1\|$  も前回の演算結果を利用できる。従って、今回新たに演算するのは  $\langle \mathbf{u}_k \cdot \mathbf{u}_i \rangle = \langle \mathbf{u}_1 \cdot \mathbf{u}_i \rangle$  の部分である。

【0101】以上を前提として、第2基底の探索は以下の演算処理を行う。即ち、ステップS41では  $k=1$  により  $P_1 = \langle \mathbf{d} \cdot \mathbf{u}_1 \rangle$ 、 $L_1 = \|\mathbf{u}_1\|^2$  を保持する。これらは上記ステップS22、S23で求めた結果を利用できる。なお、添え字の「1」は第1基底  $\langle \mathbf{u}_1 \rangle$  を指すインデックスカウンタ  $i$  の内容であり、これは上記ステッ

ブS 2 7の処理により配列  $[I_k]$  に保持されている。  
ステップS 4 2では (1 9) 式の演算を行い、結果をレジスタ  $\eta$ ,  $\kappa$  に格納する。

【0 1 0 2】

【数1 9】

$$\eta = \frac{1}{\sqrt{L_k}} \quad \kappa = P_k \eta \quad (19)$$

【0 1 0 3】ステップS 4 3では後述の内積演算  $\langle u_1 \cdot u_1 \rangle$  の前処理として  $\langle u_1 \rangle$  の第1 6成分を残りの各成分の値から差し引いた1 5次元ベクトル  $\langle w_1 \rangle$  を求める。ステップS 4 4では  $i = 0 \sim (N-1)$  につき内積  $\langle w_k \cdot u_i \rangle$   $\eta$  を求め、これらを配列  $[Q_i]$  に格納する。ステップS 4 5では  $i = 0 \sim (N-1)$  につき  $(P_i - \kappa Q_i)$  を求め、これらを配列  $[P_i]$  に格納する。ここで、右辺の  $P_i$  は上記ステップS 2 2の演算結果であり、更にこのステップS 4 5の演算結果をステップS 2 2の配列  $[P_i]$  に上書き保存することで、配列  $[P_i]$  の内容は過去の演算結果を反映して逐次更新される。ステップS 4 6では  $i = 0 \sim (N-1)$  につき  $(L_i - Q_i^2)$  を求め、これらを配列  $[L_i]$  に格納 (上書き) する。ここで、右辺の  $L_i$  は上記ステップS 2 3の演算結果であり、更にこのステップS 4 6の演算結果をステップS 2 3の配列  $[L_i]$  に上書き保存することで、配列  $[L_i]$  の内容も過去の演算結果を反映して逐次更新される。以上による  $h_i$  の繰り返し演算は最終的に (2 0) 式で表せる。

【0 1 0 4】

【数2 0】

$$h_i = \frac{(P_i - \kappa Q_i)^2}{L_i - Q_i^2} = \frac{P_i^2}{L_i} \quad (20)$$

【0 1 0 5】ステップS 4 7では  $h_i$  の最大値を保持するレジスタ  $E = 0$ , 基底候補ベクトル  $\langle u_i \rangle$  のインデックスカウンタ  $i = 0$  に夫々初期化し、かつ基底数カウンタ  $k$  に+1する。この時点で  $k = 2$  となる。

【0 1 0 6】ステップS 4 8では  $h_i = P_i^2 / L_i$  を求め

る。ステップS 4 9では  $h_i > E$  か否かを判別する。  $h_i > E$  の場合はステップS 5 0で  $E$  を  $h_i$  で更新し、その時の  $i$  を配列  $[I_k]$  ( $k = 2$ ) に保持する。また  $h_i > E$  でない場合は上記ステップS 5 0の処理をスキップする。

【0 1 0 7】ステップS 5 1では  $i$  に+1し、更にステップS 5 2では  $i \geq N$  か否かを判別する。  $i \geq N$  でない場合はステップS 4 8に戻り、次の  $h_i$  につき上記同様の最大値探索処理を行う。以下、同様にして進み、やがて、  $i \geq N$  になると全ネストブロックの探索終了である。この時、上記配列  $[I_k]$  ( $k = 2$ ) には  $h_i$  を最大とするような第2基底ベクトル  $\langle u_2 \rangle$  のインデクス値  $i$  が保持されている。

【0 1 0 8】ステップS 5 3では第2基底ベクトル  $\langle u_2 \rangle$  を  $\langle v_1 \rangle$  に正規直交化して正規化基底ベクトル  $\langle v_2 \rangle$  となし、これを配列  $[V_k]$  ( $k = 2$ ) に格納する。またスカラー係数  $\alpha_2$  ( $\langle d_1 \rangle$  の  $\langle v_2 \rangle$  への射影) を求め、これを配列  $[A_k]$  ( $k = 2$ ) に格納する。このように基底ベクトル  $\langle u_2 \rangle$  の正規直交化とスカラー係数  $\alpha_2$  の演算は上記探索結果について1 回行えばよく、これによってAOT処理の大幅な軽量化と高速化が図られる。以下も同様である。

【0 1 0 9】ステップS 5 4では残差ベクトル  $\langle d_1 \rangle$  を第2基底で近似後の残差ベクトル  $\langle d_2 \rangle = \langle d_1 \rangle - \alpha_2 \langle v_2 \rangle$  により更新する。ステップS 5 5では新たな残差ベクトルの二乗ノルム  $e = \|\langle d_2 \rangle\|^2$  を求め、更にステップS 5 6では  $e \leq Z$  か否かを判別する。  $e \leq Z$  の場合はこの段階でAOT処理を終了し、また  $e \leq Z$  でない場合は第3基底の探索処理を行う。

【0 1 1 0】図6は第3基底の探索処理を示している。この処理の説明前に該処理を効率よく行うために行った計算上の工夫を説明する。即ち、通常なら第3基底は残差ベクトル  $\langle d_2 \rangle$  との差の二乗ノルム  $e_i$  を最小とする直交ベクトル  $\langle u_i' \rangle$  として求められるが、この関係式を更に展開すると (2 1) 式が得られる。

【0 1 1 1】

【数2 1】

$$\begin{aligned} e_i &= \left\| \langle d_2 \rangle - \frac{\langle d_2 \cdot u_i' \rangle}{\|u_i'\|^2} u_i' \right\|^2 \\ &= \|d_2\|^2 - 2 \frac{\langle d_2 \cdot u_i' \rangle^2}{\|u_i'\|^2} + \frac{\langle d_2 \cdot u_i' \rangle^2}{\|u_i'\|^4} \|u_i'\|^2 \\ &= \|d_2\|^2 - \frac{\langle d_2 \cdot u_i' \rangle^2}{\|u_i'\|^2} \end{aligned} \quad (21)$$

【0 1 1 2】ここで、直交ベクトル  $\langle u_i' \rangle$  は第3基底の候補ベクトル  $\langle u_i \rangle$  を第1, 第2の正規化基底ベクトル  $\langle v_1 \rangle$ ,  $\langle v_2 \rangle$  に直交化したものである。

【0 1 1 3】

【数2 2】

$$\mathbf{u}_i' = \mathbf{u}_i - \langle \mathbf{u}_i, \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{u}_i, \mathbf{v}_2 \rangle \mathbf{v}_2 \quad (22)$$

【0114】同様にして上記(21)式右辺第1項の  $\|\mathbf{d}_2\|^2 > 0$  は基底候補によらず一定であるから、同右辺第2項を最大にするような直交ベクトル  $\langle \mathbf{u}_i' \rangle$  が第3

基底となり得る。この右辺第2項を  $h_i$  とおく。

【0115】

【数23】

$$h_i = \frac{\langle \mathbf{d}_2, \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} \quad (23)$$

【0116】更に  $h_i$  分子の直交ベクトル  $\langle \mathbf{u}_i' \rangle$  を基底候補ベクトル  $\langle \mathbf{u}_i \rangle$  で表すと、 $h_i$  分子は(24)式で表せる。

【0117】

【数24】

$$\begin{aligned} \langle \mathbf{d}_2, \mathbf{u}_i' \rangle^2 &= \langle \mathbf{d}_2, \mathbf{u}_i - \langle \mathbf{u}_i, \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{u}_i, \mathbf{v}_2 \rangle \mathbf{v}_2 \rangle^2 \\ &= (\langle \mathbf{d}_2, \mathbf{u}_i \rangle - \langle \mathbf{d}_2, \mathbf{v}_1 \rangle \langle \mathbf{u}_i, \mathbf{v}_1 \rangle - \langle \mathbf{d}_2, \mathbf{v}_2 \rangle \langle \mathbf{u}_i, \mathbf{v}_2 \rangle)^2 \\ &= \langle \mathbf{d}_2, \mathbf{u}_i \rangle^2 \quad \because \langle \mathbf{d}_2, \mathbf{v}_1 \rangle = 0 \quad \langle \mathbf{d}_2, \mathbf{v}_2 \rangle = 0 \end{aligned} \quad (24)$$

【0118】更に(24)式の残差ベクトル  $\langle \mathbf{d}_2 \rangle$  を最初の残差ベクトル  $\langle \mathbf{d} \rangle$  で表すと、 $h_i$  分子は(25)式で表せる。

【0119】

【数25】

$$\begin{aligned} \langle \mathbf{d}_2, \mathbf{u}_i \rangle^2 &= \langle \mathbf{d} - \langle \mathbf{d}, \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{d}, \mathbf{v}_2 \rangle \mathbf{v}_2, \mathbf{u}_i \rangle^2 \\ &= (\langle \mathbf{d}, \mathbf{u}_i \rangle - \langle \mathbf{d}, \mathbf{v}_1 \rangle \langle \mathbf{v}_1, \mathbf{u}_i \rangle - \langle \mathbf{d}, \mathbf{v}_2 \rangle \langle \mathbf{v}_2, \mathbf{u}_i \rangle)^2 \\ &= \left( \langle \mathbf{d}, \mathbf{u}_i \rangle - \frac{\langle \mathbf{d}, \mathbf{u}_1 \rangle \langle \mathbf{u}_1, \mathbf{u}_i \rangle}{\|\mathbf{u}_1\| \|\mathbf{u}_i\|} - \frac{\langle \mathbf{d}, \mathbf{u}_2 \rangle \langle \mathbf{u}_2, \mathbf{u}_i \rangle}{\|\mathbf{u}_2\| \|\mathbf{u}_i\|} \right)^2 \end{aligned} \quad (25)$$

【0120】同様にして  $h_i$  分母を変形すると(26)式で表せる。

【0121】

【数26】

$$\begin{aligned} \|\mathbf{u}_i'\|^2 &= \|\mathbf{u}_i - \langle \mathbf{u}_i, \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{u}_i, \mathbf{v}_2 \rangle \mathbf{v}_2\|^2 \\ &= \|\mathbf{u}_i\|^2 - \langle \mathbf{u}_i, \mathbf{v}_1 \rangle^2 - \langle \mathbf{u}_i, \mathbf{v}_2 \rangle^2 \end{aligned} \quad (26)$$

【0122】以上を(23)式の  $h_i$  に代入すると(27)式が得られる。

【0123】

【数27】

$$h_i = \frac{(\langle \mathbf{d}, \mathbf{u}_i \rangle - \langle \mathbf{d}, \mathbf{v}_1 \rangle \langle \mathbf{v}_1, \mathbf{u}_i \rangle - \langle \mathbf{d}, \mathbf{v}_2 \rangle \langle \mathbf{v}_2, \mathbf{u}_i \rangle)^2}{\|\mathbf{u}_i'\|^2 = \|\mathbf{u}_i\|^2 - \langle \mathbf{u}_i, \mathbf{v}_1 \rangle^2 - \langle \mathbf{u}_i, \mathbf{v}_2 \rangle^2}$$

【0124】ところで、(27)式分子/分母の各第2項までは既に計算されており、これらには(28)式の関係がある。

【0125】

【数28】

$$P_i = \langle \mathbf{d}, \mathbf{u}_i \rangle - \langle \mathbf{d}, \mathbf{v}_1 \rangle \langle \mathbf{v}_1, \mathbf{u}_i \rangle \quad (28-1)$$

$$L_i = \|\mathbf{u}_i\|^2 - \langle \mathbf{u}_i, \mathbf{v}_1 \rangle^2 \quad (28-2)$$

【0126】従って、 $h_i$  の演算は上記(18-2)式に習って最終的に(29)式で表せる。

【0127】

【数29】

$$h_i = \frac{\left( P_i - \frac{P_k \langle \mathbf{v}_k, \mathbf{u}_i \rangle}{\sqrt{L_k}} \right)^2}{L_i - \left( \frac{\langle \mathbf{v}_k, \mathbf{u}_i \rangle}{\sqrt{L_k}} \right)^2} \quad (29)$$

【0128】(29)式は内積  $\langle \mathbf{u}_k, \mathbf{u}_i \rangle$  が  $\langle \mathbf{v}_k, \mathbf{u}_i \rangle$  になっていることを除き、上記(18-2)式と同じ形をしている。従って、これ以降の各基底は図5と同様のルーティンを再帰的に使用することで効率よく求まる。

【0129】以上を前提として、第3基底以降の探索は以下の演算処理を行う。即ち、ステップS61では  $k=2$  により  $P_2 = \langle \mathbf{d}_1, \mathbf{u}_2 \rangle$  ,  $L_2 = \|\mathbf{u}_2\|^2$  を保持す

る。ステップS 6 2では(30)式の演算を行い、結果をレジスタ $\eta$ 、 $\kappa$ に格納する。

【0130】

【数30】

$$\eta = \frac{1}{\sqrt{L_k}} \quad \kappa = P_k \eta \quad (30)$$

【0131】ステップS 6 3では後述の内積演算 $\langle v_2 \cdot u_1 \rangle$ の前処理として $\langle v_2 \rangle$ の第16成分を残りの各成分の値から差し引いた15次元ベクトル $\langle w_2 \rangle$ を求める。但し、 $\langle v_2 \rangle$ の各成分は整数値ではないので、このままでは内積演算を実数型で行う必要が生じる。これを避けるために、予め $\langle v_2 \rangle$ (即ち、 $\langle w_2 \rangle$ )の各成分に定数 $a$ を掛けて整数にしておく。

【0132】ステップS 6 4では $i=0 \sim (N-1)$ につき内積 $(\langle w_2 \cdot u_i \rangle \eta / a)$ を求め、これらを配列 $[Q_i]$ に格納(上書)する。この時、各演算結果を定数 $a$ で割ることにより、位を戻す。ステップS 6 5では $i=0 \sim (N-1)$ につき $(P_i - \kappa Q_i)$ を求め、これらを配列 $[P_i]$ に格納(上書)する。ステップS 6 6では $i=0 \sim (N-1)$ につき $(L_i - Q_i^2)$ を求め、これらを配列 $[L_i]$ に格納(上書)する。以上により上記(29)式の演算は(31)式で表せる。

【0133】

【数31】

$$h_i = \frac{(P_i - \kappa Q_i)^2}{L_i - Q_i^2} = \frac{P_i^2}{L_i} \quad (31)$$

【0134】ステップS 6 7では $h_i$ の最大値を保持するレジスタ $E=0$ 、基底候補ベクトル $\langle u_i \rangle$ のインデックスカウンタ $i=0$ に夫々初期化し、かつ基底数カウンタ $k$ に+1する。この時点で $k=3$ となる。

【0135】ステップS 6 8では $h_i = P_i^2 / L_i$ を求める。ステップS 6 9では $h_i > E$ か否かを判別する。 $h_i > E$ の場合はステップS 7 0で $E$ を $h_i$ で更新し、その時の $i$ を配列 $[I_k]$ ( $k=3$ )に保持する。また $h_i > E$ でない場合は上記ステップS 7 0の処理をスキップする。

【0136】ステップS 7 1では $i$ に+1し、更にステップS 7 2では $i \geq N$ か否かを判別する。 $i \geq N$ でない場合はステップS 6 8に戻り、次の $h_i$ につき上記同様の最大値探索処理を行う。以下、同様にして進み、やがて、 $i \geq N$ になると全ネストブロックの探索終了である。この時、上記配列 $[I_k]$ ( $k=3$ )には $h_i$ を最大とするような第3基底ベクトル $\langle u_3 \rangle$ のインデクス値 $i$ が保持されている。

【0137】ステップS 7 3では第3基底ベクトル $\langle u_3 \rangle$ を $\langle v_1 \rangle$ 、 $\langle v_2 \rangle$ に直交化かつ正規化して正規化基底ベクトル $\langle v_3 \rangle$ となし、これを配列 $[V_k]$ に格納する。またスカラー係数 $\alpha_3$ ( $\langle d_2 \rangle$ の $\langle v_3 \rangle$ への射影)を求め、これを配列 $[A_k]$ に格納する。

【0138】ステップS 7 4では残差ベクトル $\langle d_2 \rangle$ を第3基底で近似後の残差ベクトル $\langle d_3 \rangle = \langle d_2 \rangle - \alpha_3 \langle v_3 \rangle$ により更新する。ステップS 7 5では新たな残差ベクトルの二乗ノルム $e = \|\langle d_3 \rangle\|^2$ を求め、更にステップS 7 6では $e \leq Z$ か否かを判別する。 $e \leq Z$ の場合はこの段階でAOT処理を終了し、また $e \leq Z$ でない場合はステップS 6 1に戻り、第4基底以降の前処理及び探索処理を行う。なお、図示しないが、好ましくは、例えば上記ステップS 7 6の次に $k \geq 4$ か否かの判別処理を設け、 $k \geq 4$ の場合はこのAOT処理を抜けるようにする。

【0139】以上により、AOT処理の大幅な軽量化、かつ高速化が可能となり、演算時間は実測比で従来の1/3~1/10に短縮された。

【0140】図2を参照し、AOT3 2からは $\alpha_k$ 、 $\langle v_k \rangle$ ( $k=1 \sim m$ )の組が得られ、これらの一次結合により残差ベクトル $\langle d_j \rangle$ を許容誤差 $Z$ 以内に近似できる。更に、係数変換部3 3は $\alpha_k$ 、 $\langle v_k \rangle$ ( $k=1 \sim m$ )の組を $\beta_k$ 、 $\langle u_k \rangle$ ( $k=1 \sim m$ )の組に変換すべく、以下(従来と同様)の方法により展開係数 $\beta_k$ を求める。即ち、今、基底候補ベクトル $\langle u_k \rangle$ 、展開係数 $\beta_k$ 、正規化基底ベクトル $\langle v_k \rangle$ 、スカラー係数 $\alpha_k$ の各行列を(32)式とおく時に、

【0141】

【数32】

$$\begin{aligned} U &= [u_1, u_2, \dots, u_m] & B &= \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_m \end{bmatrix} \\ V &= [v_1, v_2, \dots, v_m] & A &= \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_m \end{bmatrix} \end{aligned} \quad (32)$$

【0142】これらを(33)式で関係つける。

【0143】

【数33】

【0144】これを行列Bについて解くためには、まず行列Uを正方行列に変換すべく、両辺に行列Uの転置行列 $U^T$ を左側から掛ける。

【0145】

【数34】

$$\begin{aligned}
 U^T U &= \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{nk} \end{bmatrix} [u_1, u_2, \dots, u_{nk}] \\
 &= \begin{bmatrix} u_1 \cdot u_1 & u_1 \cdot u_2 & \cdots & u_1 \cdot u_{nk} \\ u_2 \cdot u_1 & u_2 \cdot u_2 & \cdots & u_2 \cdot u_{nk} \\ \vdots & \vdots & \ddots & \vdots \\ u_{nk} \cdot u_1 & u_{nk} \cdot u_2 & \cdots & u_{nk} \cdot u_{nk} \end{bmatrix} \quad (35)
 \end{aligned}$$

【0148】ここで $\langle u_i \cdot u_j \rangle$ は内積を表し、かつ $\langle u_i \cdot u_j \rangle = \langle u_j \cdot u_i \rangle$ であるから、対角要素に対して対称な正方行列が得られ、かつ $\langle u_i \rangle$ と $\langle u_j \rangle$ とが異なるから、逆行列が存在する。そこで、更に両辺の

$$U^T U B = U^T V A \quad (34)$$

【0146】この行列 $(U^T U)$ は、(35)式の様に展開され、

【0147】

【数35】

左側から行列 $(U^T U)$ の逆行列 $(U^T U)^{-1}$ を掛けることで、(36)式が得られ、 $\beta_k$ が求まる。

【0149】

【数36】

$$(U^T U)^{-1} U^T U B = B = (U^T U)^{-1} U^T V A \quad (36)$$

【0150】このように正規直交基底系 $\alpha_k$ 、 $\langle v_k \rangle$  ( $k=1 \sim m$ )の組を非直交基底系 $\beta_k$ 、 $\langle u_k \rangle$  ( $k=1 \sim m$ )の組に変換することにより、復号側では各基底候補ベクトル $\langle u_k \rangle$ を一々直交化する必要は無く、夫々に $\beta_k$ を掛けて加算することにより残差ベクトル $\langle d_j \rangle$ を近似できる。従って、復号処理を簡単かつ高速に行える。次に展開係数 $\beta_k$ の圧縮符号処理を説明する。

【0151】図9は実施の形態による展開係数符号処理のイメージ図である。図9(a)において、生成された $\beta_1 \sim \beta_4$ からノルム(大きさ)を抽出する。図9(b)において、ノルムを例えば昇順( $\beta_3, \beta_2, \beta_4, \beta_1$ )に並べ換え、前方(最初は0)から順に差分( $\Delta\beta_3, \Delta\beta_2, \Delta\beta_4, \Delta\beta_1$ )を求める。図9(c)において、係数残差( $\Delta\beta_3, \Delta\beta_2, \Delta\beta_4, \Delta\beta_1$ )をその下位2ビットと上位ビットとに分離し、上位ビットをハフマン符号化する。

【0152】この例では $\Delta\beta_3$ と( $\Delta\beta_2 = \Delta\beta_4 = \Delta\beta_1$ )との2組の値が発生しており、よってハフマン符号では発生頻度の高い( $\Delta\beta_2, \Delta\beta_4, \Delta\beta_1$ )にはビット数の少ない符号が割り振られ、また発生頻度の低い $\Delta\beta_3$ にはビット数の多い符号が割り振られる。従って、展開係数 $\beta_k$ の圧縮符号化が可能となる。しかも、係数残差 $\Delta\beta_k$ の上位ビットをハフマン符号化する構成により、下位ビットの端数分が切り離されることになり、よって上位ビットでは図示の如く $\Delta\beta_2 = \Delta\beta_4 = \Delta\beta_1$ となるような可能性が高い。

【0153】また、残差 $\Delta\beta_k$ の下位2ビットは正負の符号ビットと共に対応する基底ベクトル $\langle u_k \rangle$ のインデックス情報(13ビット=0~8191)と共に2バイト固定長符号エリアにパッキングされ、固定長符号として出力される。これらの符号の出力順は $\Delta\beta_3, \Delta\beta_2, \Delta\beta_4, \Delta\beta_1$ (即ち、 $u_3, u_2, u_4, u_1$ )の順である。

【0154】図9(d)において、復号側では各符号を $u_3, u_2, u_4, u_1$ の順で入力し、夫々から係数 $\Delta\beta_3, \Delta\beta_2, \Delta\beta_4, \Delta\beta_1$ を分離する。更に最初の $\Delta\beta_3$ から $\beta_3$ を復号し、該 $\beta_3$ に $\Delta\beta_2$ を加えて $\beta_2$ を復号し、該 $\beta_2$ に $\Delta\beta_4$ を加えて $\beta_4$ を復号し、そして、該 $\beta_4$ に $\Delta\beta_1$ を加えて $\beta_1$ を復号する。 $\beta_k \langle u_k \rangle$ はこれらの和(一次結合)をとって機能するものであるから、これらの順序は問題ではない。

【0155】なお、上記ノルムを昇順に並べ換え、前方(最初は0)から順に差分を求めたが、逆にノルムを降順に並べ換え、後方(最初は0)から順に差分を求めても良い。

【0156】以下、符号部34による符号処理を説明する。DPCMの予測残差 $\Delta DC_{J,1}$ については量子化係数 $Q(Z)$ で量子化すると共に、 $\Delta DC_{J,1} = 0$ の場合のみランレングスを考慮し、予測残差 $\Delta DC_{J,1}$ 及びランレングスを夫々独立にハフマン符号化する。基底数 $k$ は、 $k=0$ の場合のみランレングスを考慮し、基底数 $k$ 及びランレングスを夫々独立にハフマン符号化する。



係数残差 $\Delta\beta_k$ の上位ビットは定数 $Q$ （例えば8）で量子化した商をハフマン符号化する。また基底ベクトル $\langle u_k \rangle$ のコード情報 $i$ （＝13ビット）に展開係数 $\beta_k$ の符号ビット及び係数残差 $\Delta\beta_k$ の下位2ビットを詰めて計16ビットの固定長符号となし、これらは残差 $\Delta\beta_k$ の昇順（又は降順）に詰めて送られる。全体としては画素ブロック単位で出現順に詰めて符号列を構成する。必要なら画素ブロックの切り替わりを示すための符号EOBを書き込む。

【0157】図10は実施の形態による画像復号装置のブロック図で、上記図2の画像符号装置に対応したものである。図において、41はハフマン等による復号部、42は注目画素 $DC_j$ を含む周囲の $DC$ 値 $DC_j'$ から交流成分を含むターゲットブロック $\langle R_j \rangle$ を推定する交流成分予測部、43は復号基底系 $\beta_k \langle u_k \rangle$ （ $k=1\sim m$ ）に基づき近似残差ベクトル $\langle d_j \rangle$ を再生する残差ベクトル再生部、44は復号ブロック $\langle R_j \rangle$ に基づきターゲットブロック $\langle R_j \rangle$ を再生する $R_j$ 再生部、45は再生画像を記憶する再生画像メモリ、46は復号 $DC$ 値をIDPCM復号するIDPCM部、47は復号 $DC$ 画像を記憶する $DC$ 画像メモリ、48は図2と同様の $DC$ ネスト生成部、49は $DC$ ネストを記憶する $DC$ ネストメモリ、50は $DC$ ネストからダウンサンプルされた選択ブロック $\langle U_k \rangle$ を保持する選択ブロックバッファ、51は $\langle U_k \rangle$ に $\beta_k$ を乗算する乗算器、52、53は $\beta_k \langle U_k \rangle$ （ $k=1\sim m$ ）の累積加算部、54は累積加算結果のブロック平均値 $A_j$ を求める平均器、55は累積加算結果からブロック平均値 $A_j$ を分離する減算器、56は再生近似残差ベクトル $\langle d_j \rangle$ を保持する近似ベクトルバッファ、57はターゲットブロック $\langle R_j \rangle$ の再生 $DC$ 値 $DC_j$ に再生近似残差ベクトル $\langle d_j \rangle$ を加算する加算器である。

【0158】図11は実施の形態による画像復号処理のフローチャートである。ステップS101では画像符号データを読み込む。ステップS102では図2と同様のIDPCM法により $Y$ 、 $U$ 、 $V$ の各 $DC$ 値を解凍（復号）し、 $DC$ 画像を再生する。ステップS103では $Y$ 成分の $DC$ 画像から $DC$ ネストを生成する。この時、上記図7で示した如く、各 $DC$ 画素値 $DC_j$ の下位4ビットがマスク（＝0）され、各 $DC$ ネスト画素値 $N_j$ となる。なお、 $DC$ 画像の切り出し位置等の情報は別途に受け取る。ステップS104では原画像メモリ45及び $DC$ 画像メモリ47に対するインデクスカウンタ $j$ 、 $J$ を共に0に初期化する。

【0159】ステップS105では1ブロック画像分の符号データを入力する。ステップS106では基底数 $k=0$ か否かを判別する。 $k=0$ の場合はステップS114で後述する交流成分予測法によりターゲットブロック $\langle R_j \rangle$ を再生する。また $k \neq 0$ の場合は更にステップS107で $1 \leq k \leq 4$ か否かを判別する。

【0160】 $1 \leq k \leq 4$ の場合はステップS112で残差ベクトル $\langle d_j \rangle$ を逆量子化する。本実施の形態では予め $DC$ ネストの下位4ビットがマスク（＝0）されているため、各選択ブロック $\langle U_k \rangle$ に直接 $\beta_k$ を掛けてこれらを累積加算し、累積加算結果からそのブロック平均値 $A_j$ を1回だけ分離することで残差ベクトル $\langle d_j \rangle$ が一挙に得られる。よって復号処理が高速化される。ステップS113では得られた残差ベクトル $\langle d_j \rangle$ に対応する $DC$ 値 $DC_j$ を加算する。

【0161】また $1 \leq k \leq 4$ でない場合はステップS108でターゲットブロック $\langle R_j \rangle$ の復号データよりターゲットブロック $\langle R_j \rangle$ を直接再生する。こうして、上記何れかの方法により $4 \times 4$ 画素のターゲットブロック $\langle R_j \rangle$ が再生された。ステップS109では再生されたターゲットブロック $\langle R_j \rangle$ を再生画像メモリ45に格納する。

【0162】ステップS110ではカウンタ $j$ 、 $J$ に夫々+1し、更にステップS111では $i \geq M$ （全画素ブロック数）か否かを判別する。 $i \geq M$ でない場合はステップS105に戻り、次のブロック画像符号データにつき上記同様の復号・再生処理を行う。以下同様にして進み、やがて、ステップS111の判別で $j \geq M$ になると、1画像分の復号処理を終了する。

【0163】図12は実施の形態における交流成分予測のイメージ図で、公知の予測法を採用できる。図12

(A)は段階的交流成分予測法を示しており、以下に内容を概説する。その第1段階では注目ブロック $S$ 上の各サブブロック $S_1 \sim S_4$ を該 $S$ を含む周囲4ブロック（ $U$ 、 $R$ 、 $B$ 、 $L$ ）の各 $DC$ 値から次式により推定する。

【0164】

$$S_1 = S + (U + L - B - R) / 8$$

$$S_2 = S + (U + R - B - L) / 8$$

$$S_3 = S + (B + L - U - R) / 8$$

$$S_4 = S + (B + R - U - L) / 8$$

同様にして、この第1段階目では $U_1 \sim U_4$ 、 $L_1 \sim L_4$ 、 $R_1 \sim R_4$ 、 $B_1 \sim B_4$ 等が推定される。更に、その第2段階では上記方法を再帰的に使用することで、 $S_1$ 上の4画素 $P_1 \sim P_4$ を次式により推定する。

【0165】

$$P_1 = S_1 + (U_3 + L_2 - S_3 - S_2) / 8$$

$$P_2 = S_1 + (U_3 + S_2 - S_3 - L_2) / 8$$

$$P_3 = S_1 + (S_3 + L_2 - U_3 - S_2) / 8$$

$$P_4 = S_1 + (S_3 + S_2 - U_3 - L_2) / 8$$

$S_2 \sim S_4$ 上の各4画素 $P_1 \sim P_4$ についても同様である。このような2段階処理によりターゲットブロック $\langle R_j \rangle$ が再生される。

【0166】図13(B)は本件出願人による既提案の非段階的交流成分予測法を示しており、注目ブロック $S$ を含む周囲4ブロック（ $U$ 、 $R$ 、 $B$ 、 $L$ ）の各 $DC$ 値か

ら各サブブロック  $S_1 \sim S_4$  における各4画素  $P_1 \sim P_4$  を一挙に推定する。以下内容を概説する。まず  $S_1$  上の4画素  $P_1 \sim P_4$  を求める場合は、 $S_2 \approx S_3 \approx S$ 、 $U_3 \approx U$ 、 $L_2 \approx L$  の各近似を行う。この近似を上記  $S_1$  上の  $P_1$  の式に適用すると、

$$P_1 = S_1 + (U_3 + L_2 - S_3 - S_2) / 8 \\ = S_1 + (U + L - S - S) / 8$$

が得られる。更にこの式に上記  $S_1$  の式、 $S_1 = S + (U + L - B - R) / 8$  を代入すると、 $S_1$  上の  $P_1$  は最終的に、

$$P_1 = S + (2U + 2L - 2S - B - R) / 8$$

$$P_2 = S_1 + (U_3 + S_2 - S_3 - L_2) / 8 \\ = S_1 + (U + S - S - L) / 8$$

が得られる。更にこの式に上記  $S_1$  の式、 $S_1 = S + (U + L - B - R) / 8$  を代入すると、 $S_1$  上の  $P_2$  は最終的に、

$$P_2 = S + (2U - B - R) / 8$$

$$P_3 = S_1 + (S_3 + L_2 - U_3 - S_2) / 8 \\ = S_1 + (S + L - U - S) / 8$$

が得られる。更にこの式に上記  $S_1$  の式、 $S_1 = S + (U + L - B - R) / 8$  を代入すると、 $S_1$  上の  $P_3$  は最終的に、

$$P_3 = S + (2L - B - R) / 8$$

$$P_4 = S_1 + (S_3 + S_2 - U_3 - L_2) / 8 \\ = S_1 + (S + S - U - L) / 8$$

が得られる。更にこの式に上記  $S_1$  の式、 $S_1 = S + (U + L - B - R) / 8$  を代入すると、 $S_1$  上の  $P_4$  は最終的に、

$$P_4 = S + (2S - B - R) / 8$$

で表せる。従って、 $S_1$  上の4画素  $P_1 \sim P_4$  は、  
 $P_1 = S + (2U + 2L - 2S - B - R) / 8$   
 $P_2 = S + (2U - B - R) / 8$   
 $P_3 = S + (2L - B - R) / 8$   
 $P_4 = S + (2S - B - R) / 8$

により非段階的に一挙に求まる。 $S_2 \sim S_4$  上の各4画素  $P_1 \sim P_4$  についても同様である。

【0167】なお、上記実施の形態を具体的数値例を伴って説明したが本発明がこれらに限定されないことは明らかである。

【0168】また、上記本発明に好適なる実施の形態を述べたが、本発明思想を逸脱しない範囲内で各部の構成、制御、処理及びこれらの組合せの様々な変更が行えることは言うまでも無い。

【0169】

【発明の効果】以上述べた如く本発明によれば、DCネストの改良により高画質が得られ、またAOT演算の工夫により高速符号化が得られた。従って、HVQ方式の

高画質化、高速符号化に寄与するところが極めて大きい。

【図面の簡単な説明】

【図1】本発明の原理を説明する図である。

【図2】実施の形態による画像符号装置のブロック図である。

【図3】実施の形態による画像符号（メイン）処理のフローチャートである。

【図4】実施の形態による適応的直交変換処理のフローチャート（1）である。

【図5】実施の形態による適応的直交変換処理のフローチャート（2）である。

【図6】実施の形態による適応的直交変換処理のフローチャート（3）である。

【図7】実施の形態によるDCネストの説明図（1）である。

【図8】実施の形態によるDCネストの説明図（2）である。

【図9】実施の形態による展開係数符号処理のイメージ図である。

【図10】実施の形態による画像復号装置のブロック図である。

【図11】実施の形態による画像復号処理のフローチャートである。

【図12】実施の形態における交流成分予測のイメージ図である。

【図13】従来の画像符号装置のブロック図である。

【図14】従来の適応的直交変換処理のフローチャートである。

【図15】従来の適応的直交変換処理のイメージ図である。

【図16】従来の平均値分離処理のイメージ図である。

【符号の説明】

- 11 原画像メモリ
- 12 DC値生成部
- 13 差分PCM符号部（DPCM）
- 14 逆DPCM符号部（IDPCM）
- 15 DC画像メモリ
- 16 DCネスト生成部
- 17 DCネストメモリ
- 18 減算器
- 19 残差ベクトルバッファ
- 20 抽出部ブロックバッファ
- 21 平均器
- 22 減算器
- 23 候補ベクトルバッファ
- 24 適応的直交変換処理部（AOT）
- 25 係数変換部
- 26 符号部
- 31 DCネスト生成部

32 適応的直交変換処理部 (AOT)  
 33 係数変換部

34 符号部

【図1】

本発明の原理を説明する図

